WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Monday, December 06, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ	
☐	L38	L34 and (multiplexing or multiplexed)	21
☐	L37	L34 and l9	0
☐	L36	L34 and l22	0
☐	L35	L34 and l30	0
☐	L34	19991215	80
☐	L33	motion near3 shape near3 texture	206
☐	L32	motion and shape and texture	11975
☐	L31	L30 and motion and shape	0
☐	L30	19991215	7
☐	L29	((video adj2 objec) or VO) near8 (compress or conpressed or compressing)	53
☐	L28	((video adj2 objec) or VO) near8 (compress or conpressed or compressing) near8 stream	0
☐	L27	((video adj2 objec) or VO) near8 (compress or conpressed or compressing) near8 stream near8 (shape or motion) near8 (packet)	0
☐	L26	((video adj2 objec) or VO) near8 (compress or conpressed or compressing) near8 stream near8 (classify or classifying or classified or priority or importance) near8 (packet)	0
☐	L25	((video adj2 objec) or VO) near8 (compress or conpressed or compressing) near8 stream near8 (classify or classifying or classified) near8 (packet)	0
☐	L24	L22 and l18	7
☐	L23	L22 and l8	0
☐	L22	19991215	265
☐	L21	(((differential or differentiation or differentiated) adj service) or DiffServ or Diff-Serv or Intserv or Int-Serv)	1608
☐	L20	L19 and l11	0
☐	L19	L18 and (marker or marking)	7
☐	L18	L17 and l15	7
☐	L17	priority adj class adj queue	38
☐	L16	L15 and l8	0
☐	L15	19991215	235
☐	L14	L13	1539
☐	L13	(((differential or differentiated) adj service) or DiffServ or Diff-Serv)	1539

☐	L12	L11 and (((differential or differentiated) adj service) or DiffServ or Diff-Serv)	0
☐	L11	L10 and (bit adj rate)	11
☐	L10	L9 and video	12
☐	L9	19991215	13
☐	L8	(media or multimedia or (multi adj media)) near8 (scalable adj2 (coding or transmission or compressed or compression))	52
	DB=USPT; PLUR=YES; OP=ADJ		
☐	L7	L6 NOT l5	15
☐	L6	L4 and ((media or multimedia) near5 processing)	16
☐	L5	L4 and (multimedia or media) and multiplexing	8
☐	L4	L2 or l1	238
☐	L3	L2 and l1	0
☐	L2	(el-hady or (el adj hady) or el-Hady or El-Hady).xp.	98
☐	L1	(el-hady or (el adj hady) or el-Hady or El-Hady).xa.	140

END OF SEARCH HISTORY

L9: Entry 1 of 13 File: USPT Sep 7, 2004

DOCUMENT-IDENTIFIER: US 6788740 B1
TITLE: System and method for encoding and decoding enhancement layer data using base layer quantization data

Application Filing Date (1):
19991001

Brief Summary Text (5):
Scalable video coding is a desirable feature for many multimedia applications and services that are used in systems employing decoders with a wide range of processing power. Scalability allows processors with low computational power to decode only a subset of the scalable video stream. Another use of scalable video is in environments with a variable transmission bandwidth. In those environments, receivers with low-access bandwidth receive, and consequently decode, only a subset of the scalable video stream, where the amount of that subset is proportional to the available bandwidth.

L9: Entry 2 of 13 File: USPT Aug 24, 2004

DOCUMENT-IDENTIFIER: US 6782132 B1
TITLE: Video coding and reconstruction apparatus and methods

Application Filing Date (1):
19990811

Brief Summary Text (15):
Another example is that conventional "data adding/layering" (e.g. prediction error, scalability, etc.) hinders coding efficiency. Such often data-intensive additions might well result in excessive bit-rate, which excess must then be contained through quality-degrading methods such as quantization. Thus, conventional scalable coding is rarely utilized, and it is unlikely that high-definition media (e.g. HDTV), while ostensibly supported, can be provided at its full quality potential within available bandwidth. Other applications, such as video conferencing, are also adversely affected by these and other standard coding deficiencies.

Previous Doc Next Doc Go to Doc#

L9: Entry 8 of 13 File: USPT Jul 24, 2001

DOCUMENT-IDENTIFIER: US 6266817 B1
TITLE: Decoder for a software-implemented end-to-end scalable video delivery system

Application Filing Date (1):
19980420

Brief Summary Text (10):
Video information is extremely storage intensive, and compression is necessary during storage and transmission. Although scalable compression would be beneficial, especially for browsing in multimedia video sources, existing compression systems do not provide desired properties for scalable compression. By scalable compression it is meant that a full dynamic range of spatial and temporal resolutions should be provided on a single embedded video stream that is output by the server over the network(s). Acceptable software-based scalable techniques are not found in the prior art. For example, the MPEG-2 compression standard offers limited extent scalability, but lacks sufficient dynamic range of bandwidth, is costly to implement in software, and uses variable length codes that require additional error correction support.

Detailed Description Text (24):
Scalable compression according to the present invention is also important for image browsing, multimedia applications, transcoding to different formats, and embedded television standards. By prioritizing packets comprising the embedded stream, congestion due to contention for network bandwidth, central processor unit ("CPU") cycles, etc., in the dynamic environment of general purpose computing systems can be overcome by intelligently dropping less important packets from the transmitted embedded stream.

Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L9: Entry 13 of 13 File: EPAB Nov 26, 1998

DOCUMENT-IDENTIFIER: WO 9853613 A1
TITLE: APPARATUS, METHOD AND COMPUTER READABLE MEDIUM FOR SCALABLE CODING OF VIDEO INFORMATION

Application Date (1):
19980421

Previous Doc Next Doc Go to Doc#

L24: Entry 1 of 7 File: PGPB Jul 25, 2002

DOCUMENT-IDENTIFIER: US 20020099854 A1
TITLE: TRANSMISSION CONTROL PROTOCOL/INTERNET PROTOCOL (TCP/IP) PACKET-CENTRIC WIRELESS POINT TO MULTI-POINT (PTMP) TRANSMISSION SYSTEM ARCHITECTURE

Application Filing Date:
19990709

Detail Description Paragraph:
[0092] Currently, there are several methods that can be used in wireline network devices to implement differentiated service classes. Example methods include traffic shaping, admission control, IP precedence, and differential congestion management. It is desired that an IP-centric wireless broadband access system use all of these methods to differentiate traffic into classes of service, to map these classes of service against a QoS matrix, and thereby to simplify the operation and administration of the QoS mechanism.

Detail Description Paragraph:
[0100] The use of queue management as the primary QoS mechanism in providing QoS-based differentiated services is a simple and straight forward method for wireless broadband systems. However, wireless systems are usually more bandwidth constrained and therefore more sensitive to delay than their wireline counterparts. For this reason, it is desirable that QoS-based differentiated services be provided with mechanisms that go beyond what simple queuing can do. However, some queuing can still be required, and the different queuing methods are now discussed.

Detail Description Paragraph:
[0330] The present invention provides support of DiffServ and RSVP/int-serv by providing: (1) support of RFC 2474 and 2475; (2) DiffServ in the core of Internet; (3) RSVP/int-serv for hosts and edge networks; (4) admission control capability for DiffServ compatibility; (5) differentiated services (DSs) (a field marking supported for use by DiffServ, and translation into a wireless base station 302 resource allocation); and (6) support for binding of multiple end-to-end sessions to one tunnel session.

Detail Description Paragraph:
[0391] Block diagram 800 lists an exemplary set of priorities 812 used by downlink flow scheduler 604 to place received data packets into priority class queues. Listed are the following set of example priorities: latency-sensitive UDP prority 812a, high priority 812b, intermediate priority 812c, initial hypertext transfer protocol (HTTP) screens priority 812d, latency-neutral priority 812e, file transfer protocol (FTP), simple mail transfer protocol (SMTP) and other e-mail traffic priority 812f and low priority 812g. Persons skilled in the art will recognize that many different priority classes are possible, depending upon the QoS requirements of the end-users. Latency-sensitive UDP priority data can refer to data that has the highest priority because it is sensitive to jitter (i.e., time synchronization is important) and latency (i.e., the amount of time passage between IP data flows in reverse directions). High priority 812b can refer to, e.g., premium VPN service, and a high priority SLA service. Intermediate priority 812c can refer to, e.g., a value VPN service level and an intermediate level SLA service. HTTP screens

priority 812d can refer to the download of HTTP data, for example, an initial HTTP screen, which is important for making an Internet user feel as if he has a great deal of bandwidth available for his Internet session. Latency-neutral priority 812e can refer to data that is neutral to latency, such as, e.g., e-mail traffic. FTP, SMTP priority 812f data includes data that is insensitive to latency and jitter, but requires a large amount of bandwidth to be downloaded accurately because of the size of a transmission. Finally, low priority data 812g can refer to data that can be transmitted over a long period of time, as when one network device transmits its status information to another network device on a 24 hour basis.

Detail Description Paragraph:
[0404] SLA-based prioritization can provide a valuable means for a telecommunications provider to provide differentiated services to a variety of customers. For example, it is possible that low priority traffic from a subscriber who has purchased a premium SLA service agreement, can be scheduled at a higher priority than high priority traffic from a subscriber which has only signed up for a value level or low cost SLA service priority.

Detail Description Paragraph:
[0410] It would be apparent to those skilled in the art that other packet header fields could be useful in identifying an IP flow. The fields have been given by way of example and are not intended to be an exhaustive list of useful packet header fields. Other fields, such as, e.g., fields from IP v6 relating to differentiated services (DIFF SERV) could also be useful to IP flow analyzer 602 and 632 of wireless base station 302.

Detail Description Paragraph:
[0480] Once the type source application has been determined by packet header information or by another means, such as direct application identification, then control passes from module 1524 to module 1532 of the packet characterization component 1504. In order to identify the type of source application of the IP flow, any type of service (TOS) or differentiated service (DiffServ) field can also be analyzed.

Detail Description Paragraph:
[0489] For the new IP flow, control passes to module 1542 from module 1536 of the packet characterization component 1504. In module 1542 the packet is classified into a QoS class by performing a table lookup into IP flow QoS class table module 1544, where the types of QoS classes are stored depending on the QoS requirements for packets. Similar IP flows, (i.e., IP flows having similar QoS requirements) can be grouped together in module 1542. In classifying packets and IP flows, QoS class groupings, any DiffServ priority markings, and any TOS priority markings can be taken into account. From the module 1542, control passes to module 1548 of IP flow presentation component 1508.

Detail Description Paragraph:
[0526] Relevant fields can include, e.g., source, destination, type of service (TOS) and differentiated service (DiffServ) markings, if any exist.

Previous Doc Next Doc Go to Doc#

L19: Entry 1 of 7 File: PGPB Jul 25, 2002

DOCUMENT-IDENTIFIER: US 20020099854 A1
TITLE: TRANSMISSION CONTROL PROTOCOL/INTERNET PROTOCOL (TCP/IP) PACKET-CENTRIC WIRELESS POINT TO MULTI-POINT (PTMP) TRANSMISSION SYSTEM ARCHITECTURE

Application Filing Date:
19990709

Detail Description Paragraph:
[0092] Currently, there are several methods that can be used in wireline network devices to implement differentiated service classes. Example methods include traffic shaping, admission control, IP precedence, and differential congestion management. It is desired that an IP-centric wireless broadband access system use all of these methods to differentiate traffic into classes of service, to map these classes of service against a QoS matrix, and thereby to simplify the operation and administration of the QoS mechanism.

Detail Description Paragraph:
[0100] The use of queue management as the primary QoS mechanism in providing QoS-based differentiated services is a simple and straight forward method for wireless broadband systems. However, wireless systems are usually more bandwidth constrained and therefore more sensitive to delay than their wireline counterparts. For this reason, it is desirable that QoS-based differentiated services be provided with mechanisms that go beyond what simple queuing can do. However, some queuing can still be required, and the different queuing methods are now discussed.

Detail Description Paragraph:
[0330] The present invention provides support of DiffServ and RSVP/int-serv by providing: (1) support of RFC 2474 and 2475; (2) DiffServ in the core of Internet; (3) RSVP/int-serv for hosts and edge networks; (4) admission control capability for DiffServ compatibility; (5) differentiated services (DSs) (a field marking supported for use by DiffServ, and translation into a wireless base station 302 resource allocation); and (6) support for binding of multiple end-to-end sessions to one tunnel session.

Detail Description Paragraph:
[0391] Block diagram 800 lists an exemplary set of priorities 812 used by downlink flow scheduler 604 to place received data packets into priority class queues. Listed are the following set of example priorities: latency-sensitive UDP prority 812a, high priority 812b, intermediate priority 812c, initial hypertext transfer protocol (HTTP) screens priority 812d, latency-neutral priority 812e, file transfer protocol (FTP), simple mail transfer protocol (SMTP) and other e-mail traffic priority 812f and low priority 812g. Persons skilled in the art will recognize that many different priority classes are possible, depending upon the QoS requirements of the end-users. Latency-sensitive UDP priority data can refer to data that has the highest priority because it is sensitive to jitter (i.e., time synchronization is important) and latency (i.e., the amount of time passage between IP data flows in reverse directions). High priority 812b can refer to, e.g., premium VPN service, and a high priority SLA service. Intermediate priority 812c can refer to, e.g., a value VPN service level and an intermediate level SLA service. HTTP screens

priority 812d can refer to the download of HTTP data, for example, an initial HTTP screen, which is important for making an Internet user feel as if he has a great deal of bandwidth available for his Internet session. Latency-neutral priority 812e can refer to data that is neutral to latency, such as, e.g., e-mail traffic. FTP, SMTP priority 812f data includes data that is insensitive to latency and jitter, but requires a large amount of bandwidth to be downloaded accurately because of the size of a transmission. Finally, low priority data 812g can refer to data that can be transmitted over a long period of time, as when one network device transmits its status information to another network device on a 24 hour basis.

Detail Description Paragraph:
[0404] SLA-based prioritization can provide a valuable means for a telecommunications provider to provide differentiated services to a variety of customers. For example, it is possible that low priority traffic from a subscriber who has purchased a premium SLA service agreement, can be scheduled at a higher priority than high priority traffic from a subscriber which has only signed up for a value level or low cost SLA service priority.

Detail Description Paragraph:
[0410] It would be apparent to those skilled in the art that other packet header fields could be useful in identifying an IP flow. The fields have been given by way of example and are not intended to be an exhaustive list of useful packet header fields. Other fields, such as, e.g., fields from IP v6 relating to differentiated services (DIFF SERV) could also be useful to IP flow analyzer 602 and 632 of wireless base station 302.

Detail Description Paragraph:
[0480] Once the type source application has been determined by packet header information or by another means, such as direct application identification, then control passes from module 1524 to module 1532 of the packet characterization component 1504. In order to identify the type of source application of the IP flow, any type of service (TOS) or differentiated service (DiffServ) field can also be analyzed.

Detail Description Paragraph:
[0489] For the new IP flow, control passes to module 1542 from module 1536 of the packet characterization component 1504. In module 1542 the packet is classified into a QoS class by performing a table lookup into IP flow QoS class table module 1544, where the types of QoS classes are stored depending on the QoS requirements for packets. Similar IP flows, (i.e., IP flows having similar QoS requirements) can be grouped together in module 1542. In classifying packets and IP flows, QoS class groupings, any DiffServ priority markings, and any TOS priority markings can be taken into account. From the module 1542, control passes to module 1548 of IP flow presentation component 1508.

Detail Description Paragraph:
[0526] Relevant fields can include, e.g., source, destination, type of service (TOS) and differentiated service (DiffServ) markings, if any exist.

Detail Description Paragraph:
[0618] In one embodiment, DEN provides policy-based network management, IPsec compatible network security, and IPsec based VPNs. The DEN of the wireless base station 302 is planned to be common information model (CIM) 3.0 compatible (once the specification is finalized). The wireless base station 302 can provide native DEN support and supports directory based DEN QoS mechanisms including reservation model (i.e. RSVP, per-flow queuing), and precedence/priority/differentiated model (i.e. packet marking). Wireless base station 302 can plan support of DEN network policy QoS, and until DEN is complete, can support internal QoS and network extensions.

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L19: Entry 7 of 7 File: USPT Sep 17, 2002

DOCUMENT-IDENTIFIER: US 6452915 B1
TITLE: IP-flow classification in a wireless point to multi-point (PTMP) transmission system

Abstract Text (1):
An IP flow classification system is used in a wireless telecommunications system. The IP flow classification system groups IP flows in a packet-centric wireless point to multi-point telecommunications system. The classification system includes: a wireless base station coupled to a first data network; one or more host workstations coupled to the first data network; one or more subscriber customer premise equipment (CPE) stations in wireless communication with the wireless base station over a shared bandwidth using a packet-centric protocol; and one or more subscriber workstations coupled to each of the subscriber CPE stations over a second network; a resource allocation device optimizes end-user quality of service (QoS) and allocates shared bandwidth among the subscriber CPE stations; an analyzing and scheduling device analyzes and schedules internet protocol (IP) flow over the shared wireless bandwidth. The analyzing device includes the above IP flow classifier that classifies the IP flow. The classifier can include a device for associating a packet of an existing IP flow with the IP flow. The classifier can include a QoS grouping device that groups a packet of a new IP flow into a QoS class grouping. The QoS grouping device can include a determining device that determines and takes into account QoS class groupings for the IP flow. The QoS grouping device can include an optional differentiated services (Diff Serv) device that takes into account an optional Diff Servs field priority marking for the IP flow.

Application Filing Date (1):
19990709

Brief Summary Text (2):
The following applications of common assignee contain common disclosure: U.S. patent application Ser. No. 09/349,477 entitled "Transmission Control Protocol/Internet Protocol (TCP/IP) Packet-Centric Wireless Point to Multi-Point (PtMP) Transmission System Architecture," filed Jul. 9, 1999, U.S. patent application Ser. No. 09/349,480 entitled "Quality of Service (QoS)--Aware Wireless Point to Multi-Point (PtMP) Transmission System Architecture," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/350,126 entitled "Method for Providing Dynamic Bandwidth Allocation Based on IP-Flow Characteristics in a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/350,118 entitled "Method for Providing for Quality of Service (QoS)--Based Handling of IP-Flows in a Wireless Point to Multi-Point Transmission System," filed Jul. 9, 1999, U.S. patent application Ser. No. 09/347,356 entitled "IP-Flow Identification in a Wireless Point to Multi-Point Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/350,150 entitled "IP-Flow Characterization in a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,476 entitled "IP-Flow Prioritization in a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/350,170 entitled "Method of Operation for Providing for Service Level Agreement (SLA) Based Prioritization in a

Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,481 entitled "Method for Transmission Control Protocol (TCP) Rate Control With Link-Layer Acknowledgments in a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/350,159 entitled "Transmission Control Protocol/Internet Protocol (TCP/IP)--Centric QoS Aware Media Access Control (MAC) Layer in a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/347,857 entitled "Use of Priority-Based Scheduling for the Optimization of Latency and Jitter Sensitive IP Flows in a Wireless Point to Multi-Point Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,475 entitled "Time Division Multiple Access/Time Division Duplex (TDMA/TDD) Access Method for a Wireless Point to Multi-Point Transmission System," filed Jul. 9, 1999. U.S. patent application entitled Ser. No. 09/349,483 "Reservation Based Prioritization Method for Wireless Transmission of Latency and Jitter Sensitive IP-Flows in a Wireless Point to Multi-Point Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,479 entitled "Translation of Internet-Prioritized Internet Protocol (IP)--Flows into Wireless System Resource Allocations in a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application entitled Ser. No. 09/350,162 "Method of Operation for the Integration of Differentiated services (Diff-serv) Marked IP-Flows into a Quality of Service (QoS) Priorities in a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,975 entitled "Method for the Recognition and Operation of Virtual Private Networks (VPNs) over a Wireless Point to Multi-Point (PtMP) Transmission System," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/350,173 entitled "Time Division Multiple Access/Time Division Duplex (TDMA/TDD) Transmission Media Access Control (MAC) Air Frame," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,482 entitled "Application--Aware, Quality of Service (QoS) Sensitive, Media Access Control (MAC) Layer," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,478 entitled "Transmission Control Protocol/Internet Protocol (TCP/IP) Packet-Centric Wireless Point to Point (PtP) Transmission System Architecture," filed Jul. 9, 1999. U.S. patent application Ser. No. 09/349,474 entitled "Transmission Control Protocol/Internet Protocol (TCP/IP) Packet-Centric Cable Point to Multi-Point (PtMP) Transmission System Architecture," filed Jul. 9, 1999.

Search Forms
Search Results
Help
User Searches
Preferences
Logout

Brief Summary Text (16):
In one embodiment, the classifier includes a means for associating a packet of an existing IP flow with the IP flow. The classifier can include a QoS grouping device that groups a packet of a new IP flow into a QoS class grouping. The QoS grouping device can include a determining device that determines and takes into account QoS class groupings for the IP flow. The QoS grouping device can include an optional differentiated services (Diff Serv) device that takes into account an optional Diff Servs field priority marking for the IP flow. The QoS grouping device can also include an optional type of service (TOS) device that takes into account any optional type of service (TOS) field priority marking for said IP flow.

Detailed Description Text (39):
Currently, there are several methods that can be used in wireline network devices to implement differentiated service classes. Example methods include traffic shaping, admission control, IP precedence, and differential congestion management. It is desired that an IP-centric wireless broadband access system use all of these methods to differentiate traffic into classes of service, to map these classes of service against a QoS matrix, and thereby to simplify the operation and administration of the QoS mechanism.

Detailed Description Text (47):
The use of queue management as the primary QoS mechanism in providing QoS-based differentiated services is a simple and straight forward method for wireless broadband systems. However, wireless systems are usually more bandwidth constrained and therefore more sensitive to delay than their wireline counterparts. For this

reason, it is desirable that QoS-based differentiated services be provided with mechanisms that go beyond what simple queuing can do. However, some queuing can still be required, and the different queuing methods are now discussed.

Detailed Description Text (277):
The present invention provides support of DiffServ and RSVP/int-serv by providing: (1) support of RFC 2474 and 2475; (2) DiffServ in the core of Internet; (3) RSVP/int-serv for hosts and edge networks; (4) admission control capability for DiffServ compatibility; (5) differentiated services (DSs) (a field marking supported for use by DiffServ, and translation into a wireless base station 302 resource allocation); and (6) support for binding of multiple end-to-end sessions to one tunnel session.

Detailed Description Text (338):
Block diagram 800 lists an exemplary set of priorities 812 used by downlink flow scheduler 604 to place received data packets into priority class queues. Listed are the following set of example priorities: latency-sensitive UDP prority 812a, high priority 812b, intermediate priority 812c, initial hypertext transfer protocol (HTTP) screens priority 812d, latency-neutral priority 812e, file transfer protocol (FTP), simple mail transfer protocol (SMTP) and other e-mail traffic priority 812f and low priority 812g. Persons skilled in the art will recognize that many different priority classes are possible, depending upon the QoS requirements of the end-users. Latency-sensitive UDP priority data can refer to data that has the highest priority because it is sensitive to jitter (i.e., time synchronization is important) and latency (i.e., the amount of time passage between IP data flows in reverse directions). High priority 812b can refer to, e.g., premium VPN service, and a high priority SLA service. Intermediate priority 812c can refer to, e.g., a value VPN service level and an intermediate level SLA service. HTTP screens priority 812d can refer to the download of HTTP data, for example, an initial HTTP screen, which is important for making an Internet user feel as if he has a great deal of bandwidth available for his Internet session. Latency-neutral priority 812e can refer to data that is neutral to latency, such as, e.g., e-mail traffic. FTP, SMTP priority 812f data includes data that is insensitive to latency and jitter, but requires a large amount of bandwidth to be downloaded accurately because of the size of a transmission. Finally, low priority data 812g can refer to data that can be transmitted over a long period of time, as when one network device transmits its status information to another network device on a 24 hour basis.

Detailed Description Text (351):
SLA-based prioritization can provide a valuable means for a telecommunications provider to provide differentiated services to a variety of customers. For example, it is possible that low priority traffic from a subscriber who has purchased a premium SLA service agreement, can be scheduled at a higher priority than high priority traffic from a subscriber which has only signed up for a value level or low cost SLA service priority.

Detailed Description Text (357):
It would be apparent to those skilled in the art that other packet header fields could be useful in identifying an IP flow. The fields have been given by way of example and are not intended to be an exhaustive list of useful packet header fields. Other fields, such as, e.g., fields from IP v6 relating to differentiated services (DIFF SERV) could also be useful to IP flow analyzer 602 and 632 of wireless base station 302.

Detailed Description Text (427):
Once the type source application has been determined by packet header information or by another means, such as direct application identification, then control passes from module 1524 to module 1532 of the packet characterization component 1504. In order to identify the type of source application of the IP flow, any type of service (TOS) or differentiated service (DiffServ) field can also be analyzed.

Detailed Description Text (436):
For the new IP flow, control passes to module 1542 from module 1536 of the packet characterization component 1504. In module 1542 the packet is classified into a QoS class by performing a table lookup into IP flow QoS class table module 1544, where the types of QoS classes are stored depending on the QoS requirements for packets. Similar IP flows, (i.e., IP flows having similar QoS requirements) can be grouped together in module 1542. In classifying packets and IP flows, QoS class groupings, any DiffServ priority markings, and any TOS priority markings can be taken into account. From the module 1542, control passes to module 1548 of IP flow presentation component 1508.

Detailed Description Text (473):
Relevant fields can include, e.g., source, destination, type of service (TOS) and differentiated service (DiffServ) markings, if any exist.

Detailed Description Text (566):
In one embodiment, DEN provides policy-based network management, IPsec compatible network security, and IPsec based VPNs. The DEN of the wireless base station 302 is planned to be common information model (CIM) 3.0 compatible (once the specification is finalized). The wireless base station 302 can provide native DEN support and supports directory based DEN QoS mechanisms including reservation model (i.e. RSVP, per-flow queuing), and precedence/priority/differentiated model (i.e. packet marking). Wireless base station 302 can plan support of DEN network policy QoS, and until DEN is complete, can support internal QoS and network extensions.

CLAIMS:

5. The system of claim 4, wherein said QoS grouping device comprises: optional differentiated services (Diff Serv) device that takes into account an optional Diff Servs field priority marking for said IP flow.

6. The system of claim 4, wherein said QoS grouping device comprises: optional type of service (TOS) device that takes into account any optional type of service field priority marking for said IP flow.

23. The system according to claim 20, wherein said determining means comprises at least one of: means for storing and retrieving a source application for a source address from a source application table; means for determining a source application from a type of service (TOS) packet field; and means for determining a source application from a differentiated services (DiffServ) header field.

30. The system according to claim 29, wherein said classifying means comprises: means for taking into account any optional differentiated services (Diff Serv) field priority marking for said previously classified IP flow.

31. The system according to claim 29, wherein said classifying means comprises: means for taking into account any optional type of service (TOS) field priority marking for said previously classified IP flow.

36. The system according to claim 16, wherein said prioritizer device comprises: means for taking into account any differentiated services (DiffServ) priorities for said IP flow.

47. The system according to claim 44, wherein said determining means comprises: means for determining a source application from a differentiated services (DiffServ) packet field.

67. The system according to claim 66, wherein said QoS grouping device comprises: an optional differentiated services (Diff Serv) device operative to take into

account an optional Diff Servs field priority marking for said IP flow.

68. The system according to claim 66, wherein said QoS grouping device comprises: an optional type of service (TOS) device operative to take into account any optional type of service (TOS) field priority marking for said IP flow.

77. The system according to claim 16, wherein said prioritizer device comprises: a type of service (TOS) prioritizer that prioritizes said IP flow based on a TOS marking of a packet of said IP flow.

78. The system according to claim 16, wherein said prioritizer device comprises: a differentiated services (DiffServ) prioritizer that prioritizes said IP flow based on a DiffServ marking of a packet of said IP flow.

93. The system according to claim 1, wherein said resource allocation means comprises: means for accounting for any differentiated services (DiffServ) priorities for said IP flow.

102. The system according to claim 100, wherein said IP-priority packet IP flow identification information comprises a TOS prioritizer operative to account for any optional type of service (TOS) field priority marking.

103. The system according to claim 102, wherein said type of service (TOS) field priority marking is compatible with Internet Engineering Task Force (IETF) RFC 1992b.

104. The system according to claim 103, wherein said type of service (TOS) field priority marking is compatible with IETF RFC 1349.

105. The system according to claim 104, wherein said marking comprises: a minimize delay marking; a maximize throughput marking; a maximize reliability marking; a minimize monetary cost marking; and a normal service marking.

106. The system according to claim 100, wherein said IP-priority packet header IP flow identification information comprises a DiffServ prioritizer operative to account for any optional differential service (Diff Serv) field priority marking.

107. The system according to claim 106, wherein said Diff Serv field priority marking is compatible with Internet Engineering Task Force (IETF) RFC 2474.

108. The system according to claim 106, wherein said Diff Serv field priority marking is compatible with IETF RFC 2475.

112. The system according to claim 106, wherein said RSVP marking is compatible with Internet Engineering Task Force (IETF) RFC 2205.

160. The system of claim 159, wherein said QoS grouping device comprises: optional differentiated services (Diff Serv) device that takes into account an optional Diff Servs field priority marking for said IP flow.

161. The system of claim 159, wherein said QoS grouping device comprises: optional type of service (TOS) device that takes into account any optional type of service (TOS) field priority marking for said IP flow.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L38: Entry 1 of 21 File: PGPB Feb 12, 2004

DOCUMENT-IDENTIFIER: US 20040028129 A1
TITLE: PICTURE ENCODING METHOD AND APPARATUS, PICTURE DECODING METHOD AND APPARATUS AND FURNISHING MEDIUM

Application Filing Date:
19990624

Detail Description Paragraph:
[0109] The VOP encoding unit 3n encodes the output of the VOP constructing unit 2n in accordance with the system conforming to, for example, the MPEG or H.263 standard, and outputs the resulting bitstream to a multiplexing unit 4. The multiplexing unit 4 multiplexes the bitstreams from the VOP encoding unit 31 to VOP encoding unit 3N to transmit the resulting multiplexed data over transmission routes, such as ground waves, satellite network, CATV network or the like or to record the multiplexed data on a recording medium 6, such as magnetic disc, magneto-optical disc, optical disc or on a magnetic tape. The transmission medium 5 and the recording medium 6 represent embodiments of the furnishing medium of the present invention.

Detail Description Paragraph:
[0254] In the video object plane (video-object-plane), the motion shape texture (motion_shape_texture) is read in to encode the picture texture information and the shape encoding information. It is in this video object plane that the macro-block etc is encoded. The syntax of the motion shape texture (motion-shape-texture) is roughly made up of two portions, namely a data partitioning motion shape texture (data-partitioning-motion-shape-texture) and the combined motion shape texture (combined-motion-shape-texture). The data partitioning motion shape texture (data-partitioning-motion-shape-texture) is used if the 1-bit flag indicated by the VOL is 1 and if the texture information is transmitted.

Detail Description Paragraph:
[0255] The combined motion shape texture (combined-motion-shape-texture) is used when the flag (data_partitioning) is 0 or when only the shape information is transmitted. The combined motion shape texture (combined-motion-shape-texture) is made up of one or more macro-blocks.

Detail Description Paragraph:
[0314] Returning to the syntax of the motion shape texture (motion_shape_texture), if the flag of the data partitioning (data-partitioning) is 1 and if the texture information is transmitted, the data partitioning motion shape texture (data-partitioning-motion-shap- e-texture) is encoded. This data partitioning motion shape texture (data-partitioning-motion-shape-texture) is roughly constituted by two portions, that is the data partitioning I-VOP (data_partitioning_I_VOP) and the data partitioning P-VOP (data_partitioning_P_VOP).

Previous Doc Next Doc Go to Doc#

L38: Entry 6 of 21 File: USPT May 27, 2003

DOCUMENT-IDENTIFIER: US 6571019 B1
TITLE: Apparatus and method of encoding/decoding a coded block pattern

Application Filing Date (1):
19990507

Detailed Description Text (4):
As shown in FIG. 6, the input video signal S1 is applied to object formation unit 1. The object formation unit 1 divides the input video signal S1 into objects and obtains shape information and texture information of each object. Output signals S2 are applied to object encoding unit 2 and encoded based upon the object in respective object encoders 2-0, 2-1, . . . , 2-n. Outputs S3 from the object encoders 2-0, 2-1, . . . , 2-n, namely, bitstreams of different objects, are multiplexed in multiplexer 3, and then outputs S4 are transmitted as bitstream to a receiver or storage media, not shown.

Detailed Description Text (5):
FIG. 7 is a schematic diagram of each of the object encoders forming the object encoding unit 2 depicted in FIG. 6. According to FIG. 6, the object encoding unit is made of many object encoders. The reason for representing plural object encoders is to emphasize that the encoding is performed in object units. Actually, the object encoder as shown in FIG. 7 performs the encoding as many times as the number of objects. The input signal is information on objects coded and comprises shape information and texture information. This information for a unit frame is input and processed for each frame. The overall structure is similar to that in the frame-based coding, other than a shape information encoder which is included. The shape information, motion information, and texture information is multiplexed to provide a bitstream forming an output prediction error and a result of the encoding, texture bitstream S28, is input into the multiplexer 28. At the same time, the texture information encoder 24 reconstructs the incoming prediction error signal and provides the reconstructed prediction error signal S27 to adder 27. The adder 27 sums up the reconstructed prediction error signal S27 and the motion compensation predicted signal obtained in the motion compensator 23 and stores a result in the previous reconstructed object memory 25 as a reconstructed texture information signal of the corresponding object. This stored reconstructed texture information is used during the encoding of the next frame that is subsequently input.

Detailed Description Text (8):
The multiplexer 28 receives and multiplexes the shape bitstream, motion bitstream, texture bitstream, and various overhead information necessary for decoding to generate one bitstream.

Detailed Description Text (32):
FIG. 15 shows a detailed configuration of one of the prior art object decoders at the object decoding units in the object-based video signal decoding system depicted in FIG. 14. The object decoding unit 5 is shown to have a plurality of object decoders 5-0, 5-1, . . . , 5-n in FIG. 14 to emphasize that the decoding is performed in object units. Actually, the object decoder repeatedly operates to

individually decode signals of respective objects. As shown in FIG. 15, object bitstream S6 is demultiplexed into motion bitstream S53, shape bitstream S51, and texture bitstream S52 in the demultiplexer 51 and respectively input into decoders 52-54. Shape information decoder 52 receives the shape bitstream S51 and generates reconstructed shape information S54.

Detailed Description Text (33):
This reconstructed shape information is input into motion compensator 55 and texture information decoder 53 to permit the decoding on an object basis. Namely, it is intended that only object pixels are reconstructed. The texture information decoder 53 receives the texture bitstream S52 and performs texture information decoding. The texture information decoder 53 also receives the reconstructed shape information from the shape information decoder 52 and uses the information during the texture information decoding to decode only the object. Motion information decoder 54 receives the motion bitstream S53 and decodes motion information. Motion compensator 55 performs motion compensation prediction using the motion information from the motion information decoder 54 and previous texture information from previous reconstructed object memory 56.

CLAIMS:

1. In an object-based video coding system comprising a shape information encoding unit for encoding input shape information and providing reconstructed shape information and a texture information encoding unit including a transform unit, quantizer, scanning unit, and variable length coding (VLC) unit for encoding input texture information, a coded block pattern encoding apparatus comprising: a non-transparent block counter for detecting the number of non-transparent blocks within the macroblock using said input reconstructed shape information and generating first and second switching control signals according to the detected number of non-transparent blocks; a coded block pattern decision unit for generating numerical information for reading the coded block pattern according to the number of non-transparent blocks based upon said input reconstructed shape information; a first switch for coupling information generated from said coded block pattern decision unit to one of a plurality of paths according to the first switching control signal generated from said non-transparent block counter; a VLC table memory for providing the encoded coded block pattern according to information obtained through the first switch; a second switch for coupling the coded block pattern generated from said VLC table memory to an output path according to the second switching control signal generated from said non-transparent block counter; and multiplexing means for multiplexing the coded block pattern generated from said coded block pattern decision and encoding means and texture information generated from said VLC unit to provide an output.

6. In an object-based video coding system and object based decoding system, in which the coding system comprises a shape information encoding unit for encoding input shape information and providing reconstructed shape information and a texture information encoding unit including a transform unit, a quantizer, a scanning unit, and a variable length coding (VLC) unit for encoding input texture information, and a coded block pattern encoding apparatus and, the decoding system comprises a shape information decoding unit for decoding an input shape bitstream and providing a reconstructed shape bitstream and a texture information decoding unit including a demultiplexer, a variable length decoding (VLD) unit, an inverse scanning, an inverse quantizer, and an inverse transform unit for decoding an input texture bitstream, and a coded block pattern decoding apparatus, the coding system comprising: a coding non-transparent block counter for detecting the number of non-transparent blocks within the macroblock using said input reconstructed shape information and generating first and second switching control signals according to the detected number of non-transparent blocks; a coded block pattern decision unit for generating numerical information for reading the coded block pattern according to the number of non-transparent blocks based upon said input reconstructed shape

information; a first switch for coupling information generated from said coded block pattern decision unit to one of a plurality of paths according to the first switching control signal generated from said non-transparent block counter; a VLC table memory for providing the encoded coded block pattern according to information obtained through the first switch; a second switch for coupling the coded block pattern generated from said VLC table memory to an output path according to the second switching control signal generated from said non-transparent block counter; and multiplexing means for multiplexing the coded block pattern generated from said coded block pattern decision and encoding means and texture information generated from said VLC unit to provide an output.

9. The system of claim 8, wherein the coding system further comprises: a coding non-transparent block counter for detecting the number of non-transparent blocks within the macroblock using said input reconstructed shape information and generating first and second switching control signals according to the detected number of non-transparent blocks; a coded block pattern decision unit for generating numerical information for reading the coded block pattern according to the number of non-transparent blocks based upon said input reconstructed shape information; a first switch for coupling information generated from said coded block pattern decision unit to one of a plurality of paths according to the first switching control signal generated from said non-transparent block counter; a VLC table memory for providing the encoded coded block pattern according to information obtained through the first switch; and a second switch for coupling the coded block pattern generated from said VLC table memory to an output path according to the second switching control signal generated from said non-transparent block counter; and multiplexing means for multiplexing the coded block pattern generated from said coded block pattern decision and encoding means and texture information generated from said VLC unit to provide an output.

Previous Doc Next Doc Go to Doc#

L38: Entry 8 of 21 File: USPT Nov 19, 2002

DOCUMENT-IDENTIFIER: US 6483874 B1
TITLE: Efficient motion estimation for an arbitrarily-shaped object

Abstract Text (1):
An efficient motion estimation technique for an arbitrarily-shaped video object reduces the number of searches for motion estimation for shape coding and texture coding. The technique can be easily employed in an MPEG-4 encoder for coding Video Object Planes (VOPs). In a pixel loop, an m.times.m block traverses a reference video image with the video object, and the alpha plane value of each pixel is examined. If the alpha plane values in a block are not all the same, this indicates the block overlaps the object boundary. Accordingly, a shape coding mask value for the coordinate of a reference pixel in the block, such as the top, left pixel, is set to "1". As the block traverses the reference video image, a shape coding mask that generally follows the shape of the object boundary is developed to define a shaped reference image search area. A texture coding mask is defined similarly when the blocks overlap or are inside the object. When the same search range is used for both shape and texture motion estimation, the texture coding mask can be defined by ORing the shape coding mask with any blocks inside the object but not already included in the shape coding mask.

Application Filing Date (1):
19990127

Brief Summary Text (16):
Accordingly, it would be desirable to provide an improved, more efficient shape and texture motion estimation system for digital video objects. The system should exploit the irregular boundary of the object to reduce the number of searches. The system should also be general enough to apply with any fast block matching alternative. The system should be applicable to arbitrarily-shaped video coding algorithms, such as MPEG-4.

Brief Summary Text (21):
The invention relates to an efficient motion estimation technique for an arbitrarily-shaped video object that reduces the number of searches for motion estimation for shape coding and texture coding. The invention is particularly suitable for use in an MPEG-4 encoder for coding Video Object Planes (VOPs).

Brief Summary Text (34):
When a common search range is used for both shape and texture motion estimation coding of the video object, the respective mask values may be set for each of the blocks to indicate the search region for texture motion estimation by ORing: (a) the alpha plane values of the respective reference pixels of the blocks which indicate the search region for shape motion estimation with (b) the alpha plane values of the respective reference pixels of the blocks which are inside the video object in the reference video image.

Detailed Description Text (5):
An identical figure of complexity remains in motion estimation of both shape and texture coding. In reality, an exhaustive search over the entire search area is

redundant because of the arbitrary shape. Part of the search area that covers only the background does not have any chance of being matched. Efficient motion estimation is possible for an arbitrarily-shaped object by searching only on the effective search area. This opportunity has not existed before with conventional rectangular-shaped object.

Detailed Description Text (9):
The frame 105 and VOP data from frame 115 are supplied to separate encoding functions. In particular, VOPs 117, 118 and 119 undergo shape, motion and texture encoding at encoders 137, 138 and 139, respectively. With shape coding, binary and gray scale shape information is encoded. With motion coding, the shape information is coded using motion estimation within a frame. With texture coding, a spatial transformation such as the DCT is performed to obtain transform coefficients which can be variable-length coded for compression.

Detailed Description Text (24):
Motion information (e.g., motion vectors) is provided from the motion estimation function 220 to the MUX 280, while shape information which indicates the shape of the VOP is provided from the shape coding function 210 to the MUX 280. The MUX 280 provides a corresponding multiplexed data stream to a buffer 290 for subsequent communication over a data channel.

Detailed Description Text (54):
If the same search range is used for shape and texture motion estimation, the shaded regions are the same as those of FIG. 5, with the addition of the interior regions 612, 622 and 632 of the objects 410, 420 and 430, respectively. In this case, the mask for texture coding can be obtained by ORing the mask for shape coding with the original objects (i.e., ORing FIGS. 4 and 5, respectively).

Detailed Description Text (55):
Otherwise, if a different search range is used for shape and texture motion estimation, a separate pass as described in steps 1B-3B is required to obtain the texture coding mask.

CLAIMS:

11. The method of claim 1, wherein: a common search range is used for both shape and texture motion estimation coding of said video object; and said respective mask values are set for each of said blocks to indicate the search region for texture motion estimation by ORing: (a) the alpha plane values of the respective reference pixels of the blocks which indicate the search region for shape motion estimation with (b) the alpha plane values of the respective reference pixels of the blocks which are inside said video object in said reference video image.

26. The apparatus of claim 16, wherein: a common search range is used for both shape and texture motion estimation coding of said video object; and said respective mask values are set for each of said blocks to indicate the search region for texture motion estimation by ORing: (a) the alpha plane values of the respective reference pixels of the blocks which indicate the search region for shape motion estimation with (b) the alpha plane values of the respective reference pixels of the blocks which are inside said video object in said reference video image.

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Search Forms
Search Results
Help
User Searches
Preferences
Logout

Generate Collection

L38: Entry 11 of 21 File: USPT Oct 23, 2001

DOCUMENT-IDENTIFIER: US 6307885 B1
TITLE: Device for and method of coding/decoding image information

Application Filing Date (1):
19970925

Brief Summary Text (23):
The respective VOPs defined by the VOP definition block 110 are transferred into VOP coding blocks 120a, 120b, . . . , and 120n to perform a coding by VOPs. They are then multiplexed in a multiplexer 130 and transmitted as bit streams.

Brief Summary Text (33):
The motion information estimated by the motion compensation block 121, the texture information encoded by the texture coding block 124, and the shape information encoded by the shape coding block 127 are multiplexed by a multiplexer 128, and they are transmitted as a bit stream into the multiplexer 130 as shown in FIG. 1.

Brief Summary Text (36):
The VOP encoded signal is transferred from the demultiplexer 210 into a shape decoding block 221, a motion decoding block 222, and a texture decoding block 225, decoding the shape, motion, and texture information of the VOP.

Brief Summary Text (61):
A device for coding image information in the fifth construction, which is a image encoder for splitting an image transferred from an image input into an object image and a background image that have predetermined shape information, includes: an unit block definition section receiving an image signal and defining unit blocks having predetermined shape information; a motion estimation section for estimating the motion of the image transferred from the unit block definition section in the units of a macro block; a motion compensation section for motion-compensating the motion information transferred from the motion estimation section; a subtractor for obtaining the difference value between the motion-compensated unit blocks transferred from the motion compensation section and the unit blocks defined by the unit block definition section; a texture coding section for coding the difference value and shape information transferred from the subtractor until boundary blocks are received, merging the boundary blocks by a boundary block merge technique, and variable-length-coding the merged boundary blocks according to the characteristics of the merged boundary blocks; an adder for obtaining the sum value between the motion-compensated unit blocks transferred from the motion compensation section and the interior information coded by the texture coding section; a previous reconstructed unit section detector for detecting the unit blocks of the previous image from the output of the adder, and transmitting them to the motion estimation section and the motion compensation section; a shape coding section for coding the shape information concerning the image transferred from the unit block definition section in the units of a macro block; a multiplexer for multiplexing the motion information estimated by the motion estimation section, the interior information coded by the texture coding section, and the shape information coded by the shaped coding section; and a buffer for transmitting the output of the multiplexer as bit streams.

Brief Summary Text (71):
The device for decoding image information according to the forth construction, comprises: a demultiplexer for demultiplexing a coded image signal including a multiplexed image signal variable-length-coded after a boundary block merge; a shape decoder for receiving the image signal from the demultiplexer, and decoding shape information; a motion decoding section for receiving the demultiplexed image signal from the demultiplexer, and decoding motion information; a texture decoding section for receiving the demultiplexed image signal from the demultiplexer, and decoding interior coded information; an unit block memory for storing reconstructed unit blocks; a motion compensation section for receiving the motion information from the motion decoding section, the shape information from the shape decoding section, and information concerning the previous unit block reconstructed by the unit block memory; a unit block reconstruction section for receiving the motion-compensated information from the motion compensation section, the shape information from the shape decoding section, and the interior information from the texture decoding section, and reconstructing the image in the units of an unit block; and a composition section for composing the image concerning the unit blocks reconstructed by the unit block reconstruction block.

Detailed Description Text (57):
The motion information estimated by the motion estimation section S54, the texture coded by the texture coding section S56, and the shape information coded by the shape coding section S52 are multiplexed by a multiplexer S53 and generated as bit streams by a buffer S58.

Detailed Description Text (89):
A demultiplexer R41 receives the coded image signal including the image signal processed by a BBM technique, coded by a VLC and multiplexed, demultiplexing it by unit blocks. The concept of the unit block is the same as defined for the VOP of MPEG-4.

Previous Doc Next Doc Go to Doc#

L38: Entry 15 of 21 File: USPT Feb 15, 2000

DOCUMENT-IDENTIFIER: US 6026195 A
TITLE: Motion estimation and compensation of video object planes for interlaced digital video

Application Filing Date (1):
19990428

Detailed Description Text (4):
The frame 105 and VOP data from frame 115 are supplied to separate encoding functions. In particular, VOPs 117, 118 and 119 undergo shape, motion and texture encoding at encoders 137, 138 and 139, respectively. With shape coding, binary and gray scale shape information is encoded. With motion coding, the shape information is coded using motion estimation within a frame. With texture coding, a spatial transformation such as the DCT is performed to obtain transform coefficients which can be variable-length coded for compression.

Detailed Description Text (18):
Motion information (e.g., motion vectors) is provided from the motion estimation function 220 to the MUX 280, while shape information which indicates the shape of the VOP is provided from the shape coding function 210 to the MUX 280. The MUX 280 provides a corresponding multiplexed data stream to a buffer 290 for subsequent communication over a data channel.

L38: Entry 19 of 21 File: USPT Nov 23, 1999

DOCUMENT-IDENTIFIER: US 5990957 A
TITLE: Video signal bit amount control using adaptive quantization

Application Filing Date (1):
19971104

Detailed Description Text (5):
Input video information is sequentially selected in the unit of VOPs by a VOP selector 10. A shape encoder 11 extracts shape information corresponding to contour for the selected VOP and encodes such information and outputs the result. A motion estimator 12 subtracts an input video signal from a previous reconstructed VOP output form a previous frame VOP reconstruction portion 14 to output a differential video signal. A motion compensator 13 defines characteristic values for frequency components of the information output from the motion estimator 12. A texture encoder 15 encodes a signal output from the motion compensator 13 to output texture information. A multiplexer 16 receives shape information, motion information and texture information, respectively output from the shape encoder 11, motion estimation portion 12 and texture encoder 15, and multiplexes the received information. A buffer 17 receives the multiplexed video input from the multiplexer 16, and temporarily stores and output the same.

Detailed Description Text (47):
The value obtained from the equation (3) given above is just an estimated value of the texture bit. In MPEG-1 or MPEG-2, since the processing is generally performed in the unit of frames, the bit rate is mostly composed of texture bit and motion bit values. However, if the processing must be performed in the unit of objects, like in MPEG-4, shape bits for expressing the objects are additionally necessary. Therefore, controlling the bit rate is allowed by mainly using the quantization step size of the texture bit, and the motion bit and shape bit are also taken into consideration. By adding the motion bit and shape bit to the texture bit, the entire bit rate for a VOP is given by the equation (4):

L38: Entry 20 of 21 File: USPT Nov 2, 1999

DOCUMENT-IDENTIFIER: US 5978034 A
**** See image for Certificate of Correction ****
TITLE: Moving picture encoding method and apparatus, moving picture decoding method and apparatus and recording medium

Abstract Text (1):
A picture encoding method in which a picture constituting a motion vector is divided into an object picture in which the main concern is a picture making up the moving picture and the other picture, the object picture is further divided into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object and the texture and shape pictures are respectively encoded. The motion vector of the shape picture is detected to output the motion vector of the shape picture, and the shape picture is encoded based on the motion vector of the shape picture to output encoded data of the shape picture. The motion vector of the shape picture is encoded to output encoded data of the motion vector of the shape picture, and the motion vector of the texture picture is decoded, using the motion vector of the shape picture, to output the motion vector of the texture picture. The texture picture is encoded based on the motion vector of the texture picture to output encoded data of the texture picture and the motion vector of the texture picture is encoded using the motion vector of the shape picture to output encoded data of the motion vector of the texture picture. This improves the encoding efficiency while facilitating decoding control.

Application Filing Date (1):
19980218

Brief Summary Text (13):
Since the motion compensated inter-frame prediction is effective both for encoding the texture and for encoding the shape, the motion compensated inter-frame prediction is used in the object scalable encoding for these two. Since the motion vector of the texture is correlated with that of the shape, it is practised to use the motion vector for the texture for predicting the motion vector of the shape.

Brief Summary Text (17):
The texture motion vector detected by the texture motion detector 109 is sent to a texture motion compensation unit 110 and to the texture motion vector encoder 106 for texture encoding, while also being sent to a shape motion detector 102 and to a shape motion vector encoder 105 for shape encoding. The texture motion compensation unit 110 creates a prediction texture picture from the locally decoded picture, using the texture motion vector, and enters the picture to the texture encoder 111. The texture encoder 111 encodes the input texture on the block basis. The texture motion vector encoder 106 calculates the difference between the texture motion vector and the texture motion vector of a previously encoded block to encode the resulting difference texture motion vector.

Brief Summary Text (18):
The shape entered from the shape input terminal 101 is sent to the shape motion detector 102 and to a shape encoder 104 as later explained. The shape motion detector 102 detects the amount of motion between the input shape and the locally

decoded shape picture locally decoded by the shape decoder 104, on the block basis. In detecting this shape motion vector, reference is had to the texture motion vector in order to find the motion vector having a lesser difference from the texture motion vector so as not to increase the amount of generated bits at the time of encoding the shape motion vector as later explained. The detected shape motion vector is entered to a shape motion compensation unit 103 and to the shape motion vector encoder 105 for shape encoding. The shape motion vector encoder 105 calculates a difference between the shape motion vector and the texture motion vector of the previously encoded block to encode the difference shape motion vector. The shape motion compensation unit 103 generates a prediction shape picture from the locally decoded shape picture, using the shape motion vector, to enter the produced prediction shape picture in the shape encoder 104. The shape decoder 104 encodes the input shape, based on the prediction shape picture, from one block to another.

Brief Summary Text (19):
Output signals of the shape decoder 104, shape motion vector encoder 105, texture motion vector encoder 106 and the texture encoder 111 are multiplexed by a multiplexer 107 so as to be outputted as encoded data at a code output terminal 112. This encoded data is transmitted over a communication network to a receiving side, or recorded on a recording medium for later reproduction by a reproducing device.

Brief Summary Text (20):
The encoding method for the shape motion vector and the texture motion vector is summarized. The texture motion vector is encoded as a difference from the texture motion vector of the previously detected block (difference texture motion vector). The shape motion vector is encoded as a difference from the texture motion vector of the previously encoded block, that is the directly previous block (difference shape motion vector).

Brief Summary Text (22):
The time flow of encoding of the above-mentioned shape motion vector and texture motion vector is as shown in FIG. 5. The processing of FIG. 5 is iteration from block to block. The following processing is carried out for each block.

Brief Summary Text (23):
First, at step ST101, one of the previously encoded texture motion vectors (usually, the texture motion vector lying on the left or upper side of the block being encoded) is selected, and a difference shape motion vector between the texture motion vector and the shape motion vector of the block being encoded is calculated and encoded.

Brief Summary Text (26):
The reciprocal reference between the texture motion vector and the shape motion vector is as shown in FIG. 6, from which it is seen that, in encoding the texture motion vector and the shape motion vector of the blocks B101 to B103, reference is had to the texture motion vector of the previously detected (encoded) other blocks and the differences (residuals) is encoded.

Brief Summary Text (28):
That is, in FIG. 4, the encoded data from a transmission network, received by a receiving device, not shown, or encoded data from a recording medium, reproduced by the reproducing device, are separated by a demultiplexer 122 into codes of the shape, shape motion vector, texture and the texture motion vector.

Brief Summary Text (40):
In one aspect, the present invention provides a picture encoding apparatus in which a picture constituting a moving picture is divided into an object picture in which the main concern is a picture making up the moving picture and the other picture,

the object picture is split into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object, the texture picture and the shape picture being encoded independently. The apparatus of the first aspect includes first motion vector detection means for detecting the motion vector of the shape picture and to output the motion vector of the shape picture, and first encoding means for encoding the shape picture based on the motion vector of the shape picture to output encoded data of the shape picture. The first encoding means also encodes the motion vector of the shape picture to output encoded data of the motion vector of the shape picture. The apparatus also includes second motion vector detection means for detecting the motion vector of the texture picture and for outputting the motion vector of the texture picture and second encoding means for encoding the texture picture based on the motion vector of the texture picture to output encoded data of the texture picture. The second encoding means also encodes the motion vector of the texture picture using the motion vector of the shape picture to output encoded data of the motion vector of the texture picture.

Brief Summary Text (41):
In another aspect, the present invention provides a picture decoding apparatus for decoding an encoded signal representing encoded moving picture signals, in which the encoded signal is made up of encoded data of a shape picture, encoded data of a motion vector of the shape picture, encoded data of a texture picture, and encoded data of a motion vector of the texture picture, and in which each of the encoded data is generated by dividing a picture constituting a moving picture into an object picture the main concern of which is a picture making up the moving picture and the other picture, splitting the object picture into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object, detecting the motion vector of the shape picture to output the motion vector of the shape picture, encoding the shape picture based on the motion vector of the shape picture to output encoded data of the shape picture, encoding the motion vector of the shape picture based on the motion vector of the texture picture to output encoded data of the motion vector of the shape picture, encoding the motion vector of the shape picture to output encoded data of the motion vector of the shape picture, detecting the motion vector of the texture picture to output the motion vector of the texture picture, encoding the texture picture based on the motion vector of the texture picture to output encoded data of the texture picture and by encoding the motion vector of the texture picture using the motion vector of the shape picture to output encoded data of the motion vector of the texture picture. The apparatus of the second aspect includes first decoding means for decoding encoded data of the motion vector of the shape picture to output a decoded motion vector of the shape picture and for decoding encoded data of the shape picture based on the decoded motion vector of the shape picture to output a decoded shape picture. The apparatus also includes second decoding means for decoding encoded data of the motion vector of the texture picture using the decoded motion vector of the shape picture to output a decoded motion vector of the texture picture and for decoding encoded data of the texture picture based on the decoded motion vector of the texture picture to output a decoded texture picture.

Brief Summary Text (42):
In a further aspect, the present invention provides a recording medium decodable by a decoding device, the recording medium having recorded thereon encoded signals representing encoded moving picture signals. The encoded signal is made up of encoded data of a shape picture, encoded data of a motion vector of the shape picture, encoded data of a texture picture, and encoded data of a motion vector of the texture picture. Each of the encoded data is generated by dividing a picture constituting a moving picture into an object picture in which the main concern is a picture making up the moving picture and the other picture, splitting the object picture into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object, detecting the motion

vector of the shape picture to output the motion vector of the shape picture, encoding the shape picture based on the motion vector of the shape picture to output encoded data of the shape picture, encoding the motion vector of the shape picture based on the motion vector of the texture picture to output encoded data of the motion vector of the shape picture, encoding the motion vector of the shape picture to output encoded data of the motion vector of the shape picture, detecting the motion vector of the texture picture to output the motion vector of the texture picture, encoding the texture picture based on the motion vector of the texture picture to output encoded data of the texture picture and by encoding the motion vector of the texture picture using the motion vector of the shape picture to output encoded data of the motion vector of the texture picture.

Brief Summary Text (43):
According to the present invention, the encoding efficiency may be improved and the decoding control facilitated by encoding the motion vector of the shape picture and by encoding the motion vector of the texture picture by employing the motion vector of the shape picture.

Brief Summary Text (44):
Also, according to the present invention, the difference shape motion vector, which is a difference value between the shape motion vector and the shape motion vector of the previously encoded block, and the difference texture motion vector, which is a difference value between the texture motion vector and the shape motion vector of the same block, are encoded, so that the difference vector, which is a difference value from the different sorts of the vectors of the different blocks, is not encoded, thus improving the encoding efficiency and facilitating decoding control.

Drawing Description Text (7):
FIG. 6 illustrates the sequence of finding the shape motion vector and the texture motion vector in the conventional technique.

Drawing Description Text (10):
FIG. 9 illustrates the sequence of finding the shape motion vector and the texture motion vector embodying the present invention.

Drawing Description Text (12):
FIG. 11 is a flowchart for illustrating the sequence of calculating residuals with the aid of the texture in the shape motion detection embodying the present invention.

Detailed Description Text (8):
On the other hand, the texture motion detector 9 searches the motion vector in the neighborhood of the shape motion vector as candidates for the texture motion vector in order to detect the texture motion vector. That is, a search range in which to detect the texture motion vector is set for the reference texture picture, with a position specified by the shape motion vector as the center, and the area inside the search range is searched for detecting the texture motion vector. The detected texture motion vector is entered to a texture motion vector encoder 6 and to a texture motion vector compensation unit 10. In detecting the texture motion vector, the locally decoded shape picture, as later explained, is used. That is, since the texture motion vector is detected on the block basis, the locally decoded shape picture is used and the operation of detecting the background portion is omitted, if a given block contains an edge between the human being and the background.

Detailed Description Text (10):
The texture motion vector encoder 6 calculates the difference between the input texture motion vector and the shape motion vector of the same block from the shape motion detector 2 to encode the difference texture motion vector.

Detailed Description Text (11):

Output codes of the shape encoder 4, shape motion vector encoder 5, texture motion vector encoder 6 and the texture encoder 11 are multiplexed by a multiplexer 7 so as to be outputted as an encoded signal at the code output terminal 12.

Detailed Description Text (13):
The time flow of encoding of the shape motion vector and the texture motion vector in the above-described embodiment is shown in FIG. 8. The processing of FIG. 8 is a repetition of the same processing from block to block. The particular processing in each block is as follows:

Detailed Description Text (19):
The relation of reciprocal pertinence between the texture motion vector and the shape motion vector is shown in FIG. 9, from which it is seen that respective left-side blocks of the blocks B1 to B3 are previous blocks, and the shape motion vector refers at all times in each of these blocks B1 to B3 to the left side (previous) shape motion vectors, while the texture motion vector refers at all times in each of these blocks B1 to B3 to the shape motion vector of the same block. The differences (residuals) from the referenced motion vector are encoded.

Detailed Description Text (24):
Specifically, it is assumed that a candidate of the shape motion vector and the texture motion vector is (V.sub.x, V.sub.y), the shape motion vector of the left side (previous) block is (V.sub.xo, V.sub.yo), S(i,j) is a pixel of the shape in a block for encoding, S.sub.R (i,j) is a pixel of the shape in a locally decoded block, T(i,j) is a pixel of the texture in the block for encoding and T.sub.R (i,j) is a pixel of the texture in a locally decoded block. Referring to the flowchart of FIG. 11, the sum of absolute values (evaluation value) of residuals of corresponding pixels between the shape block for encoding and the block in a search range of the locally decoded shape picture, which is a reference picture, is found. Then, processing transfers to step ST22 to find the sum of absolute values (evaluation value) of residuals of the corresponding pixels between the texture block for the shape block for encoding and the block in a search range of the locally decoded texture picture which is a reference picture corresponding to the shape search range. The value equal to a multiple equal to a constant c of the evaluation value of the texture is summed to the evaluation value of the shape as found at step ST21. The value c is usually 1/256 for the hard key and 1/2 for a soft key. The resulting sum is an updated new evaluation value D. Then, processing transfers to step ST13 where the length of the shape motion vector (sum of absolute values V.sub.x, V.sub.y) multiplied by a constant a (usually on the order of 1/4) is summed to the evaluation value D as found at step ST22. The resulting value is an updated new evaluation value D. Then, processing transfers to step ST24 to check whether or not the vector of the shape for encoding is the motion vector of the shape of the left side (previous) block. If the result is YES, processing transfers to step ST25 where a constant b (usually of the order of 5) is subtracted and the resulting value is updated as a new evaluation value D. If the result is NO, the current value is kept. The above-described sequence of operations is carried out for all blocks in the search range of the reference picture and the block for encoding to select the shape motion vector corresponding to the position minimizing the evaluation value. The difference of the present embodiment from the embodiment of FIG. 10 is that the value corresponding to a multiple of the constant c of the residuals of the texture is summed the evaluation value of the shape as found at step ST21.

Detailed Description Text (25):
In encoding the shape motion vector and the texture motion vector of the current block in the above-described embodiment of the encoding device, the difference shape motion vector, which is the difference value from the motion vector of the shape of another previously encoded block is encoded for the shape motion vector, while a difference texture motion vector, as a difference from the shape motion vector of the same block, is encoded for the texture motion vector. That is, in the

present embodiment of the moving picture encoding device, the difference vector from the different sorts of vectors in the different blocks is not encoded, thus giving an improved encoding efficiency.

Detailed Description Text (30):
Referring to FIG. 12, data entering the code input terminal 21, that is encoded data from the transmission network, received from a receiving device, not shown, or encoded data from a recording medium, obtained by a reproducing device, are separated by a demultiplexer 22 into codes of the shape, shape motion vector, texture and texture motion vector.

Detailed Description Text (32):
The texture motion vector decoder 24 decodes input codes to generate a difference texture motion vector. The shape motion vector of the same block as that of the difference texture motion vector is summed to the difference shape motion vector to decode the texture motion vector. This texture motion vector is sent to a texture motion compensation unit 28.

Detailed Description Text (38):
According to the present invention, as described above, the encoding efficiency may be improved and the decoding control facilitated by encoding the motion vector of the shape picture and by encoding the motion vector of the texture picture by employing the motion vector of the shape picture.

Detailed Description Text (39):
Also, according to the present invention, the difference shape motion vector, which is a difference value between the shape motion vector and the shape motion vector of the previously encoded block, and the difference texture motion vector, which is a difference value between the texture motion vector and the shape motion vector of the same block, are encoded, so that the difference vector, which is a difference value from the different sorts of the vectors of the different blocks, is not encoded, thus improving the encoding efficiency as well as facilitating decoding control.

CLAIMS:

1. A picture encoding apparatus in which a picture constituting a moving picture is divided into an object picture in which the main concern is a picture making up the moving picture and the other picture, said object picture is split into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object, said texture picture and the shape picture being encoded independently, comprising:

first motion vector detection means for detecting the motion vector of the shape picture and to output the motion vector of the shape picture;

first encoding means for encoding said shape picture based on the motion vector of the shape picture to output encoded data of the shape picture, said first encoding means also encoding the motion vector of said shape picture to output encoded data of the motion vector of the shape picture;

second motion vector detection means for detecting the motion vector of the texture picture and for outputting the motion vector of the texture picture; and

second encoding means for encoding said texture picture based on the motion vector of said texture picture to output encoded data of said texture picture, said second encoding means also encoding the motion vector of the texture picture using the motion vector of the shape picture to output encoded data of the motion vector of the texture picture.

3. The picture encoding apparatus as claimed in claim 2 wherein

said second motion vector detection means searches a range in the vicinity of a position specified by a motion vector of said shape picture to detect the motion vector of said texture picture.

6. A picture encoding method in which a picture constituting a moving picture is divided into an object picture in which the main concern is a picture making up the moving picture and the other picture, said object picture is split into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object, said texture picture and the shape picture being encoded independently, comprising:

a first motion vector detection step for detecting the motion vector of the shape picture and to output the motion vector of the shape picture;

a first encoding step for encoding said shape picture based on the motion vector of the shape picture to output encoded data of the shape picture, said first encoding step also encoding the motion vector of said shape picture to output encoded data of the motion vector of the shape picture;

a second motion vector detection step for detecting the motion vector of the texture picture and for outputting the motion vector of the texture picture; and

a second encoding step for encoding said texture picture based on the motion vector of said texture picture to output encoded data of said texture picture, said second encoding means also encoding the motion vector of the texture picture using the motion vector of the shape picture to output encoded data of the motion vector of the texture picture.

8. The picture encoding method as claimed in claim 7 wherein

said second motion vector detection step searches a range in the vicinity of a position specified by a motion vector of said shape picture to detect the motion vector of said texture picture.

11. A picture decoding apparatus for decoding an encoded signal representing encoded moving picture signals, in which

the encoded signal is made up of encoded data of a shape picture, encoded data of a motion vector of the shape picture, encoded data of a texture picture, and encoded data of a motion vector of the texture picture, and in which each of said encoded data is generated by dividing a picture constituting a moving picture into an object picture in which the main concern is a picture making up the moving picture and the other picture, splitting said object picture into a texture picture representing brightness and the color hue of the picture and a shape picture representing the shape of an object, detecting the motion vector of the shape picture to output the motion vector of the shape picture, encoding said shape picture based on the motion vector of the shape picture to output encoded data of the shape picture, encoding the motion vector of said shape picture based on the motion vector of the shape picture to output encoded data of the motion vector of the shape picture, encoding the motion vector of the shape picture to output encoded data of the motion vector of the shape picture, detecting the motion vector of the texture picture to output the motion vector of the texture picture, encoding said texture picture based on the motion vector of the texture picture to output encoded data of the texture picture and by encoding the motion vector of the texture picture using the motion vector of said shape picture to output encoded data of the motion vector of said texture picture, said apparatus comprising:

first decoding means for decoding encoded data of the motion vector of the shape

picture to output a decoded motion vector of the shape picture, said first decoding means also decoding encoded data of said shape picture based on the decoded motion vector of the shape picture to output a decoded shape picture; and

second decoding means for decoding encoded data of the motion vector of the texture picture using the decoded motion vector of the shape picture to output a decoded motion vector of the texture picture, said second decoding means also decoding encoded data of said texture picture based on the decoded motion vector of the texture picture to output a decoded texture picture.

12. The picture decoding apparatus as claimed in claim 11 wherein

the motion vector of said shape picture and the motion vector of said texture picture are each detected from block to block;

the encoded data of the motion vector of the shape picture being obtained by calculating a difference between the motion vector of the shape picture of a block to be encoded and the motion vector of the shape picture of the neighboring block and encoding the resulting difference shape motion vector;

the encoded data of the motion vector of the texture picture being obtained by calculating a difference between the motion vector of the texture picture of a block to be encoded and the motion vector of the shape picture of the same block and encoding the resulting difference shape motion vector;

said first decoding means executing motion compensated predictive decoding using said shape decoded picture and summing the motion vector of the difference texture of the block to be encoded to the motion vector of the shape picture of the neighboring block to restore the motion vector of the shape picture; and

said second encoding means executing motion compensated predictive decoding using said texture decoded picture and summing the motion vector of the difference texture of the block to be encoded to the motion vector of the shape picture of the neighboring block to restore the motion vector of the texture picture.

13. A picture decoding method for decoding an encoded signal representing encoded moving picture signals, in which

the encoded signal is made up of encoded data of a shape picture, encoded data of a motion vector of the shape picture, encoded data of a texture picture, and encoded data of a motion vector of the texture picture, and in which each of said encoded data is generated by dividing a picture constituting a moving picture into an object picture in which the main concern is a picture making up the moving picture and the other picture, splitting said object picture into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object, detecting the motion vector of the shape picture to output the motion vector of the shape picture, encoding said shape picture based on the motion vector of the shape picture to output encoded data of the shape picture, encoding the motion vector of the shape picture to output encoded data of the motion vector of the shape picture, detecting the motion vector of the texture picture using the motion vector of the shape picture to output the motion vector of the texture picture, encoding said texture picture based on the motion vector of the texture picture to output encoded data of the texture picture and by encoding the motion vector of the texture picture using the motion vector of said shape picture to output encoded data of the motion vector of said texture picture, said apparatus comprising:

a first decoding step for decoding encoded data of the motion vector of the shape picture to output a decoded motion vector of the shape picture, said first decoding step also decoding encoded data of said shape picture based on the decoded motion

vector of the shape picture to output a decoded shape picture; and

a second decoding step for decoding encoded data of the motion vector of the texture picture using the decoded motion vector of the shape picture to output a decoded motion vector of the texture picture, said second decoding step also decoding encoded data of said texture picture based on the decoded motion vector of the texture picture to output a decoded texture picture.

14. The picture decoding method as claimed in claim 13 wherein

the motion vector of said shape picture and the motion vector of said texture picture are each detected from block to block;

the encoded data of the motion vector of the shape picture being obtained by calculating a difference between the motion vector of the shape picture of a block to be encoded and the motion vector of the shape picture of the neighboring block and encoding the resulting difference shape motion vector;

the encoded data of the motion vector of the texture picture being obtained by calculating a difference between the motion vector of the texture picture of a block to be encoded and the motion vector of the shape picture of the same block and encoding the resulting difference shape motion vector;

said first decoding step executing motion compensated predictive decoding using said shape decoded picture and summing the motion vector of the difference texture of the block to be encoded to the motion vector of the shape picture of the neighboring block to restore the motion vector of the shape picture; and

said second encoding step executing motion compensated predictive decoding using said decoded texture picture and summing the motion vector of the difference texture of the block to be encoded to the motion vector of the shape picture of the same block to restore the motion vector of the texture picture.

15. A recording medium decodable by a decoding device, said recording medium having recorded thereon encoded signals representing encoded moving picture signals, the encoded signal is made up of encoded data of a shape picture, encoded data of a motion vector of the shape picture, encoded data of a texture picture, and encoded data of a motion vector of the texture picture, and in which each of said encoded data is generated by dividing a picture constituting a moving picture into an object picture in which the main concern is a picture making up the moving picture and the other picture, splitting said object picture into a texture picture representing brightness and color hue of the picture and a shape picture representing the shape of an object, detecting the motion vector of the shape picture to output the motion vector of the shape picture, encoding said shape picture based on the motion vector of the shape picture to output encoded data of the shape picture, encoding the motion vector of said shape picture to output encoded data of the motion vector of the shape picture, detecting the motion vector of the texture picture using the motion vector of the shape picture to output the motion vector of the texture picture, encoding said texture picture based on the motion vector of the texture picture to output encoded data of the texture picture and by encoding the motion vector of the texture picture using the motion vector of said shape picture to output encoded data of the motion vector of said texture picture.

16. The recording medium as claimed in claim 15 wherein

the motion vector of said shape picture and the motion vector of said texture picture are each detected from block to block;

the encoded data of the motion vector of the shape picture being obtained by

calculating a difference between the motion vector of the shape picture of a block to be encoded and the motion vector of the shape picture of the neighboring block and encoding the resulting difference shape motion vector;

the encoded data of the motion vector of the texture picture being obtained by calculating a difference between the motion vector of the texture picture of a block to be encoded and the motion vector of the shape picture of the same block and encoding the resulting difference shape motion vector.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L44: Entry 1 of 2 File: USPT Dec 24, 2002

DOCUMENT-IDENTIFIER: US 6499060 B1
TITLE: Media coding for loss recovery with remotely predicted data units

Application Filing Date (1):
19990312

Detailed Description Text (52):
The bit stream of the compressed video sequence includes the shape, motion and texture coded information from the shape coding, motion estimation, and texture coding modules. In addition, it includes overhead parameters that identify the type of coding used at the frame, object, and macroblock levels. Depending on the implementation, the parameter designating the data type as I, R, or P can be encoded at any of these levels. In the implementation, the encoder places a flag at the macroblock level to indicate which type of data is encoded in the macroblock. Multiplexer 544 combines and formats the shape, motion, texture, and overhead parameters into the proper syntax and outputs it to the buffer 546. The multiplexer may perform additional coding of the parameters. For example, overhead parameters and the motion vectors may be entropy coded, using a conventional entropy coding technique such as Huffman or arithmetic coding.

Detailed Description Text (54):
FIG. 6 is a block diagram illustrating a decoder for an object-based video coding method. A demultiplexer 660 receives a bit stream 662 representing a compressed video sequence and separates shapes, motion and texture encoded data on an object by object basis. The demultiplexer also includes a mode selector to indicate whether the macroblocks are I, P, or R type.

Detailed Description Text (67):
As introduced above, the encoder implementation may also prioritize the transfer of I, P, and R encoded data units with respect to the reliability of the data transfer. Preferably, the encoded stream should be prioritized such that the I unit has the highest reliability of being transferred, the R unit has the same or lower reliability as the I unit, and the P unit has the same or lower probability as the R unit. Additionally, if the encoder implements a multi-layered R encoding scheme, the priority for each level R.sub.n would decrease. For example, units of type R.sub.1, dependent on the I unit in the segment, would have the highest priority, units of type R.sub.2, dependent on unit of type R.sub.1 in the sub-segment, would have the next highest priority.

Detailed Description Text (68):
The implementation prioritizes data transfer by transmitting the higher priority data units first so that there is more time to retransmit them if they are not received. Once prioritized, the encoded data units maintain that priority level on retransmission. In particular, all higher priority data units are transmitted, including retransmissions if necessary, before lower priority data units.

Detailed Description Text (71):
The prioritization scheme works in the case of a transfer between a server and a single client, and in a multi-cast scenario where a server broadcasts the data

packets to multiple clients. The extent to which the client uses the encoded units to reconstruct the output depends on its available transmission bandwidth. In the case of a multi-cast application, the client can subscribe to the number of priority levels that it can support based on its available bandwidth. Consider an example such as FIG. 2B where the data stream is encoded with multiple layers of R units, and the units in each layer are dependent on the R unit in a lower level sent with higher priority. Also, consider the case where the frame rate of the R units designated in the input stream doubles with each level of hierarchy. In particular, the highest priority R units, R.sub.1, are inserted so as to double the frame rate relative to the I units, the next highest priority R units, R.sub.2, are inserted so as to double the frame rate relative to the R.sub.1 units, and so on. Depending on the bandwidth available to the client, it can subscribe to however many levels it wants, with each level providing better fidelity yet increasing the bandwidth requirements. By sending all units of a given priority level in a portion of the stream first, this approach sacrifices the lower priority units to make sure that the higher priority units will be received.

Previous Doc Next Doc Go to Doc#

L38: Entry 14 of 21 File: USPT May 2, 2000

DOCUMENT-IDENTIFIER: US 6057884 A
TITLE: Temporal and spatial scaleable coding for video object planes

Application Filing Date (1):
19970605

Detailed Description Text (4):
The frame 105 and VOP data from frame 115 are supplied to separate encoding functions. In particular, VOPs 117, 118 and 119 undergo shape, motion and texture encoding at encoders 137, 138 and 139, respectively. With shape coding, binary and gray scale shape information is encoded. With motion coding, the shape information is coded using motion estimation within a frame. With texture coding, a spatial transformation such as the DCT is performed to obtain transform coefficients which can be variable-length coded for compression.

Detailed Description Text (67):
In a further application of the present invention, an asynchronous transfer mode (ATM) communication technique is presented. Generally, the trend towards transmission of video signals over ATM networks is rapidly growing. This is due to the variable bit rate (VBR) nature of these networks which provides several advantages over constant bit rate (CBR) transmissions. For example, in VBR channels, an approximately constant picture quality can be achieved. Moreover, video sources in ATM networks can be statistically multiplexed, requiring a lower transmission bit rate than if they are transmitted through CBR channels since the long term average data rate of a video signal is less than the short term average due to elastic buffering in CBR systems.